Journal of Statistical Software

August 2012, Volume 50, Issue 8. *http://www.jstatsoft.org/*

Nonparametric Kernel Distribution Function Estimation with kerdiest: An R Package for Bandwidth Choice and Applications

Alejandro Quintela-del-Río
University of A Coruña

Graciela Estévez-Pérez
University of A Coruña

Abstract

The R package **kerdiest** has been designed for computing kernel estimators of the distribution function and other related functions. Because of its usefulness in real applications, the bandwidth parameter selection problem has been considered, and a cross-validation method and two of plug-in type have been implemented. Moreover, three relevant functions in nature hazards have also been programmed. The package is completed with two interesting data sets, one of geological type (a complete catalogue of the earthquakes occurring in the northwest of the Iberian Peninsula) and another containing the maximum peak flow levels of a river in the United States of America.

Keywords: nonparametric estimation, distribution function, bandwidth selection, exceedance function, mean return period, return level.

1. Introduction

Let $(x_1, x_2, \ldots, x_n)$ be a data sample of a continuous random variable X, with distribution function F and density f. A natural estimator for the distribution function F is the empirical distribution one, defined at any point x as

$$F_n(x) = n^{-1} \sum_{j=1}^{n} I_{(-\infty,x]}(x_j). \tag{1}$$

On the other hand, a well-known nonparametric estimator of the density function is the Rosenblatt-Parzen (Parzen 1962) kernel estimator $\hat{f}_h(x) = n^{-1} \sum_{j=1}^{n} K_h(x - x_j)$, where $K_h(u) = h^{-1}K(u/h)$ with K the kernel function and h the bandwidth parameter. Using the relationship between the density and the distribution function, that is, $F(y) = \int_{-\infty}^{y} f(t)dt$, it

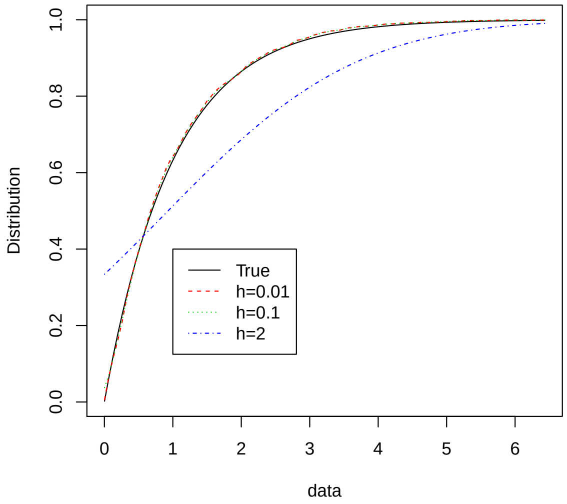

Figure 1: Effect of the bandwidth on the kernel estimator for an exponential distribution with parameter $\beta = 1$.

is easy to construct a kernel estimator for the distribution function as:

$$\hat{F}_h(x) = \int_{-\infty}^{x} \hat{f}_h(t)dt, \qquad (2)$$

which can also be written in a similar form as the kernel estimator of the density, by

$$\hat{F}_h(x) = n^{-1} \sum_{j=1}^{n} H\left(\frac{x - x_j}{h}\right), \qquad (3)$$

where $H(x) = \int_{-\infty}^{x} K(t)dt$. Some theoretical properties of the estimator $\hat{F}_h$ have been investigated, among others, by Nadaraya (1964), Reiss (1981) or Hill (1985).

As seen in Equation 3, when working with a kernel estimator of the distribution function two choices must be made: the kernel function K and the smoothing parameter or bandwidth h. The selection of K is a problem of less importance, and different functions that produce good results can be used. In practice, the choice of an efficient method for the calculation of h, for an observed data sample is a more complex problem, because of the effect of the bandwidth on the shape of the corresponding estimator. If the bandwidth is small, we will obtain an undersmoothed estimator, with high variability. On the contrary, if the value of h is big, the resulting estimator will be very smooth and farther from the function that we are trying to estimate.

An example is drawn in Figure 1, where we show three different kernel estimators using the same kernel function (the standard Gaussian density) and three different values for the bandwidth. The data sample consists of 1000 random numbers of an exponential distribution with parameter $\beta = 1$.

Despite the great number of bandwidth selection techniques in other settings, as for example in density or regression estimation (Jones, Marron, and Sheather 1996; del Río 1996), to the best of our knowledge, only two methods have been investigated in the distribution estimation context: plug-in and cross-validation. The plug-in bandwidth choice was studied, both theoretically and by simulation studies, by Altman and Leger (1995) and Polanski and Baker (2000). The least-squares cross-validation method was analyzed in Sarda (1993), but, as revealed in Altman and Leger (1995), it basically requires very large sample sizes to ensure good results. Hence, only the second approach, namely the modified cross-validation proposed in Bowman, Hall, and Prvan (1998), is of interest for implementation in a language programming and for application to real data sets.

The distribution function estimation is not only an interesting problem by itself, but also for the fact that it appears naturally in real problems of many scientific fields, such as seismology, hydrology, environmental sciences, etc. Thus, diverse methodologies, based on nonparametric ideas, have emerged for attacking statistical problems in these disciplines. In many cases, the distribution function appears to be directly linked to the *risk* term, or *nature hazard*. Scientists are interested in knowing the risk of occurrence of an earthquake of great magnitude, the probability of high wind speeds or hurricane occurrences, or the hazard of high flow levels. We cite, among others, the papers of Elsner, Jagger, and Tsonis (2006), Gomes, Arrúe, López, Sterk, and Richard (2003), Katz, Parlange, and Naveau (2002), Küchenhoff and Thamerus (1996), del Río and Francisco-Fernández (2011) and Scheitlin, Elsner, Malmstadt, Hodges, and Jagger (2010) for applications of the distribution function estimation to the different sciences mentioned in the nature hazard setting.

For the above reasons, we have implemented, in the package **kerdiest**, developed in the language R (R Development Core Team 2012), the kernel distribution function estimator, the three commented bandwidth selection procedures, and three interest functions in real applications: the *exceedance*, the *mean return period* and the *return level* functions. The package also contains two application data sets, that show the features and capabilities of the package in practice. The package is available from the Comprehensive R Archive Network at `http://CRAN.R-project.org/package=kerdiest`.

The aim of this paper is to describe this package, and also to summarize and conveniently present the associated nonparametric framework, helping interested readers to apply this kind of techniques to real situations. The structure of this paper is as follows. Section 2 details the bandwidth selection procedures implemented in the package. Section 3 introduces the utility of estimating some functions derived directly from the distribution one, and their mathematical forms are presented. Section 4 explains the structure of the package. In Sections 4 and 5, the functions and data sets of the package are described, and applications of the different features are illustrated. Section 6 is devoted to conclusions.

2. Bandwidth selection

2.1. Plug-in bandwidth selection

This method is based on considering some type of quadratic error between the true function and its estimator, such as the mean integrated squared error (MISE),

$$MISE(\hat{F}_h) = \int_{-\infty}^{+\infty} (\hat{F}_h(x) - F(x))^2 dx, \tag{4}$$

and then selecting the bandwidth minimizing an asymptotic approximation of this error. It can be proven (Altman and Leger 1995) that, under smoothness conditions,

$$\begin{aligned} MISE(\hat{F}_h) &= h^4 \int_{-\infty}^{+\infty} B_F^2(x)dx + \frac{1}{n}\int_{-\infty}^{+\infty} F(x)(1-F(x))dx - \\ &\quad -\frac{h}{n}\int_{-\infty}^{+\infty} V_F^2(x)dx + o(MISE(h)), \end{aligned} \tag{5}$$

where

$$B_F(x) = \left(\tfrac{1}{2}\right)(f'(x))^2 \left(\int_{-\infty}^{+\infty} x^2 K(x)dx\right) \quad \text{and} \quad V_F^2(x) = 2f(x)\left(\int_{-\infty}^{+\infty} xK(x)H(x)dx\right). \tag{6}$$

Therefore, the asymptotically optimal bandwidth has the form:

$$h_{AMISE}(\hat{F}_h) = Cn^{-1/3} = \left(\frac{\frac{1}{2}\int_{-\infty}^{+\infty} V_F^2(x)dx}{\int_{-\infty}^{+\infty} B_F^2(x)dx}\right)^{1/3} n^{-1/3}. \tag{7}$$

As we can see in this last equation, the optimal bandwidth is of order $n^{-1/3}$, instead of $n^{-1/5}$, that would be the optimal order in the case of the kernel nonparametric density estimation (Silverman 1986). Therefore, for large sample sizes, the optimal bandwidth for the nonparametric distribution estimator will be smaller than for the corresponding density estimator. For small values of the bandwidth, the nonparametric density estimator produces a closer estimation to the true density, but the area under the estimated curve and the X axis (measured by the distribution function estimator) will be a good estimation of the true area. However, if we choose a large value for the bandwidth, we obtain a smoother estimator, away from the true curve, and therefore the estimated values for the area will be also away from the real ones. Looking at one specific graph of the distribution function estimation, the corresponding curve is a smoother one than the density estimation. For this reason, we need to use smaller values for the bandwidth of the distribution estimator than in the density case, because we have to capture the curvature changes of the distribution function.

Because the constant C in Equation 7 depends on the kernel function and the theoretical (unknown in practice) distribution function of the data, a *plug-in* estimation of considers the bandwidth

$$\hat{h} = \hat{C}\, n^{-1/3}, \tag{8}$$

where $\hat{C}$ is estimated through the data sample. The way of obtaining $\hat{C}$ differs from one author to another.

2.2. Altman and Leger plug-in

The plug-in method developed by Altman and Leger (1995) consists in estimating nonparametrically the unknown terms in Equation 7. Using Altman and Leger's notation, Equation 7 can be written as:

$$h_{AMISE}(\hat{F}_h) = \left(\frac{\frac{1}{4}V_2}{B_3}\right)^{1/3} n^{-1/3}, \tag{9}$$

with

$$V_2 = \rho(K) \int_{-\infty}^{+\infty} [f(x)]^2 dx \quad ; \quad \rho(K) = 2 \int_{-\infty}^{+\infty} x K(x) H(x) dx \tag{10}$$

and

$$B_3 = 0.25 \left(\mu_2(K)^2\right) \int_{-\infty}^{+\infty} [f'(x)]^2 f(x) dx \quad ; \quad \mu_2(K) = \int_{-\infty}^{+\infty} x^2 K(x) dx. \tag{11}$$

So the plug-in bandwidth is

$$h_{AL} = \left(\frac{\frac{1}{4}\hat{V}_2}{\hat{B}_3} \right)^{1/3} n^{-1/3}, \tag{12}$$

where

$$\hat{V}_2 = \rho(K) \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} \frac{1}{\alpha} K\left(\frac{x_i - x_j}{\alpha} \right), \tag{13}$$

and

$$\hat{B}_3 = 0.25 \hat{D}_3(F) \left(\mu_2(K)\right)^2, \tag{14}$$

being

$$\hat{D}_3(F) = \frac{1}{n^3 \alpha_b^4} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} K_b^{'} \left(\frac{x_i - x_j}{\alpha_b} \right) K_b^{'} \left(\frac{x_i - x_k}{\alpha_b} \right). \tag{15}$$

In this last formula, $K_b^{'}$ is the derivative of a kernel function K_b (that is not necessarily equal to K). Its associated bandwidth parameter is denoted as α_b. In practice, we can choose $\alpha_b = \alpha$ and $K_b = K$. Moreover, if we use as the kernel function the Epanechnikov density, Altman and Leger (1995) prove that an optimal choice is made taking $\alpha = n^{-0.3}\hat{\sigma}(x_i)$, with $\hat{\sigma}(x_i)$ an estimate of the standard deviation of the data. A good option for this last quantity is (Silverman 1986):

$$\hat{\sigma}(x_i) = \min \left\{ \hat{s}, \frac{Q_3 - Q_1}{1.349} \right\}, \tag{16}$$

with $\hat{s}$ the sample standard deviation, and Q_1, Q_3 denoting the first and third quartile, respectively. Note that we are only using this last formula for the estimation of the standard deviation, and this is independent of the estimated function. In fact, Polanski and Baker (2000) use the same formula, because it is more suitable for non-normal densities than the sample standard deviation.

2.3. Polansky and Baker plug-in

Based also on Equation 8, the Polansky and Baker plug-in bandwidth (using their notation) can be written as

$$h_{PB} = \left(\frac{\rho(K)}{-n\mu_2^2(K)\hat{\Psi}_2(g_2)} \right)^{1/3}, \tag{17}$$

where $\rho(K)$ and $\mu_2(K)$ appear in Equation 10 and Equation 11, respectively,

$$\hat{\Psi}_r(g) = \frac{1}{n^2 g^{r+1}} \sum_{i=1}^{n} \sum_{j=1}^{n} L^{(r)} \left(\frac{x_i - x_g}{g} \right) \tag{18}$$

estimates

$$\Psi_r = \int_{-\infty}^{+\infty} f^{(r)}(x) f(x) dx, \tag{19}$$

$r \geq 2$ an even integer and

$$g_2 = \left(\frac{2L^{(2)}(0)}{-n\mu_2^2(L)\Psi_4} \right)^{1/5}. \tag{20}$$

Note that the kernel function L is not necessarily equal to K.

Based on this scheme, Polanski and Baker (2000) developed an iterative method for calculating the plug-in bandwidth, that we detail below.

Let $b > 0$ be an integer.

First step. Calculate $\hat{\Psi}_{2b+2}$ using the formula

$$\hat{\Psi}_r = \frac{(-1)^{r/2} r!}{(2\hat{\sigma}(x_i))^{r+1}(r/2)!\pi^{1/2}},$$

where $\hat{\sigma}(x_i)$ is estimated as in Equation 16.

Second step. Begin from $j = b$ to $j = 1$, calculating $\hat{\Psi}_{2j}(\hat{g}_{2j})$ where

$$\hat{g}_{2j} = \left(\frac{2L^{(2j)}(0)}{-n\mu_2(L)\hat{\Psi}_{2j+2}} \right)^{1/(2j+3)}$$

with

$$\hat{\Psi}_{2j+2} = \begin{cases} \hat{\Psi}_{2b+2} & \text{if} \quad j = b \\ \hat{\Psi}_{2j+2}(\hat{g}_{2j+2}) & \text{if} \quad j < b \end{cases}.$$

Third step. The plug-in bandwidth is

$$\hat{h} = \left(\frac{\rho(K)}{-n\mu_2^2(K)\hat{\Psi}_{2j}(\hat{g}_{2j})} \right)^{1/3}. \tag{21}$$

In practice, it is sufficient to consider $b = 2$ for most applications.

2.4. Cross-validation bandwidth selection

Applied several times in the nonparametric setting, the cross-validation procedure is based on directly estimating the function *MISE* in Equation 4, and then selecting the bandwidth to minimize this function. Sarda (1993) proposed to use

$$CV(h) = \sum_{i=1}^{n} (F_n(x_i) - \hat{F}_{-i}(x_i))^2. \tag{22}$$

In this case, the $CV(\cdot)$ function minimizes the differences between the empirical distribution function (see Equation 1) and the leave-one-out version of the distribution kernel estimator. This last estimator is defined as that using all the points except x_i:

$$\hat{F}_{-i}(x) = \frac{1}{n-1} \sum_{j \neq i} H\left(\frac{x - x_j}{h} \right). \tag{23}$$

In spite of the asymptotic optimality theorem proven in Sarda (1993), this method does not provide good results in practice. Instead, the modified cross-validation proposal of Bowman *et al.* (1998) is also asymptotically optimal and works well in simulation studies and real cases. It consists in minimizing the function

$$CV(h) = \frac{1}{n}\sum_{i=1}^{n}\int_{-\infty}^{+\infty}\Big(I(x-x_i) - \hat{F}_{-i}(x)\Big)^2\,dx, \tag{24}$$

where $I(x-x_i) = 1$ if $x - x_i \geq 0$ and 0 in any other case. Bowman *et al.* (1998) performed a simulation study comparing this method with the plug-in one of Altman and Leger. Better results are obtained, in general, with cross-validation. A drawback is the worse performance in terms of computational time, because cross-validation involves the minimization of a function of n^2 terms that is necessary to evaluate over a large enough grid of bandwidths. Furthermore, it is also precise to compute an integral term (approximating it numerically). Obviously, this is not really a drawback, for a real data situation, because the minimization process is carried out only once.

3. Associated functions

In real applications, the distribution function estimation appears as a natural problem due to the existence of related functions, that can be very useful in several contexts. One of these is the risk function or probability of exceedance. For example, in climatological studies, for a high value c, the relevance of knowing the probability of occurrence of a wind speed bigger than c is obvious. The function returning the probabilities of exceedance is simply defined as

$$R(c) = 1 - F(c). \tag{25}$$

Usually, the *exceedance function* is calculated over D time units. For instance, seismologists are interested in the probability of exceedance or the risk function over D years, that is, the probability of occurrence of an earthquake of magnitude bigger than c in D years. More exactly, let us suppose that we are observing the occurrence process of earthquakes, or extreme rainfall events, in a specific geographical area. The usual assumption is that the occurrence process is a Poisson random variable. In this case, it can be proven that the exceedance function takes the form (see e.g., Orlecka-Sikora 2008):

$$R(c, D) = 1 - \exp(-\lambda D(1 - F(c)), \tag{26}$$

and we can estimate this last function by means of

$$\hat{R}_h(c, D) = 1 - \exp(-\hat{\lambda} D(1 - \hat{F}_h(c)), \tag{27}$$

where $\hat{\lambda}$ is an estimate of the mean rate λ of the Poisson process (as e.g., the sample mean).

This definition clearly concerns other disciplines, for example, measuring concentrations of a pollutant, the flow level, etc. Directly linked with it, we also consider the *mean return period* or *recurrence interval* of a concrete value c, as an estimator of the interval of time between events of level greater than or equal to c. It can be calculated as the inverse of the probability that a level c will be exceeded in 1 period of time:

$$RT(c) = \frac{1}{\lambda(1 - F(c))}. \tag{28}$$

The last function can be estimated by means of

$$\widehat{RT}(c) = \frac{1}{\hat{\lambda}(1 - \hat{F}_h(c))}. \tag{29}$$

Finally, from the definition of quantile, we also take into account the *return level* function. For $0 < p < 1$, the *quantile* of order $1 - p$ is defined as the value x_p such that $1 - p = F(x_p)$. In many practical problems it is useful to estimate quantiles corresponding to a probability of exceedance. The $T-$return level is defined as the value of the observed variable (flow level, pollutant level, wind speed) that can be expected to be once exceeded during a $T-$period of time. This matches with the quantile or the value x_T such that the probability that the random variable will be less than it is at most $1 - (1/T)$, that is,

$$x_T = F^{-1}\left(1 - \frac{1}{T}\right) \tag{30}$$

(see e.g., Coles 2001). We can estimate this last function directly by means of the nonparametric kernel distribution estimator as:

$$RL(T) = \hat{x}_T = \hat{F}_h^{-1}\left(1 - \frac{1}{T}\right). \tag{31}$$

Some examples of the three functions described appear in Sections 5.1 and 5.2.

4. The package

We have implemented in R the estimators of the defined functions and the bandwidth selection procedures of the above sections. The package **kerdiest** contains seven functions and two application data sets. In Table 1 we can find a summary of the contents of the package.

The function `kde` corresponds to the distribution function estimator (Equation 3). Four possibilities are allowed for the kernel functions: `"e"` for the Epanechnikov function, `"n"` for the $N(0, 1)$ (Gaussian) density, `"b"` for the biweight kernel and `"t"` for the triweight kernel (see e.g., Wand and Jones 1995). Obviously, there are many possibilities that could be used

Function	Description
`kde`	The distribution function estimator, as defined in Equation 3.
`ALbw`	Plug-in bandwidth selection of Altman and Leger.
`PBbw`	Plug-in bandwidth selection of Polansky and Baker.
`CVbw`	Cross-validation bandwidth selection of Bowman, Hall and Prvan.
`ef`	Exceedance function estimator.
`mpr`	Mean return period estimator.
`rl`	Return level estimator.
Data set	**Description**
`nwip`	Data of earthquakes of the northwest of the Iberian Peninsula.
`saltriver`	Data of annual peak instantaneous flow of the Salt River near Roosevelt, AZ, USA, for 1924–2009.

Table 1: Summary of contents of the package.

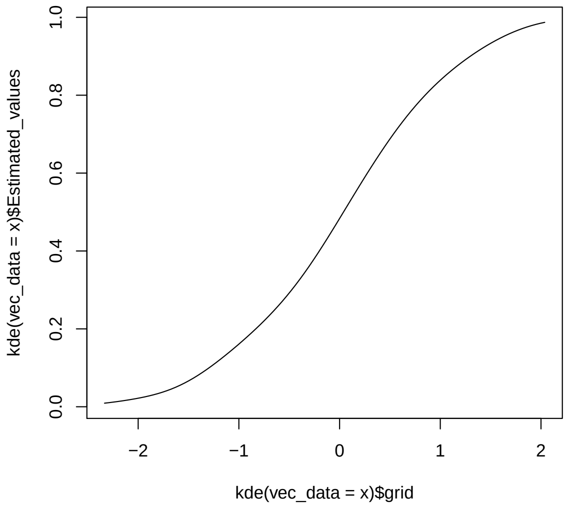

Figure 2: Distribution function estimator obtained with the function `kde`.

for the kernel function, but only these four options are applied because they are widely used in practice in the nonparametric estimation framework. Working with a data sample, from now on denoted by `vec_data`, this function enables to compute the kernel distribution function estimator over a grid of points, with a bandwidth selected by the user, but it also allows to estimate directly this parameter by the plug-in method of Polansky and Baker (`PBbw`). We have chosen this method as the automatic one because it is the fastest in computation time terms.

```
R> x <- rnorm(100)
R> kde(vec_data = x)
```

With these two last elementary lines, we generate a sample of 100 random numbers of a $N(0,1)$ distribution and estimate its distribution function. By default, the function `kde` selects a grid of 100 points in the data range. The output is a list containing the estimated values in the points of the grid, this last sequence and the Polansky and Baker plug-in bandwidth. In Figure 2 we show the estimator obtained with the code:

```
R> plot(kde(vec_data = x)$grid, kde(vec_data = x)$Estimated_values,
+    type = "l")
```

4.1. Bandwidth selection methods

The function `ALbw` provides the plug-in bandwidth of Altman and Leger. The same possibilities for the kernel function as in the function `kde` appear here. The following example computes this bandwidth for a sample of 100 random numbers of a $N(0,1)$ distribution:

```
R> x <- rnorm(100)
```

Arguments	Description
`type_kernel`	The kernel function.
`vec_data`	The data sample.
`n_pts`	The number of points used to approximate the integral term by the Simpson's rule. The default is 100 points.
`seq_bws`	The bandwidth sequence where the cross-validation score is computed.
Results	Description
`seq_bws`	The bandwidth sequence where the cross-validation score was computed.
`CVfunction`	The evaluation of the cross-validation function in the grid of bandwidths.
`bw`	The cross-validation bandwidth.

Table 2: Summary of arguments and results of `CVbw`.

```
R> h_AL <- ALbw(type_kernel = "e", x)
R> print(h_AL)
```

```
[1] 0.4257229
```

The function `PBbw` computes the plug-in bandwidth with the method of Polansky and Baker. In this case, because kernel estimators of derivatives of order greater than 2 of the distribution function are needed, we only work with the normal kernel. The value by default for the number of iterations is $b = 2$, but values up to 4 are allowed (see Section 2.3 for details). Next, we show an example:

```
R> x <- rnorm(30)
R> h_PB <- PBbw(vec_data = x, num_stage = 4)
R> print(h_PB)
```

```
[1] 0.5910396
```

The function `CVbw` computes the cross-validation bandwidth of Bowman, Hall and Prvan. The integral term in the cross-validation score (Equation 24) is calculated using the Simpson's rule (Burden and Faires 2000). This procedure approximates the integral by a sum of 100 terms, by default. Depending on the sample size, this procedure can be quite slow. An example is the following:

```
R>  x <- rnorm(30)
R> CVbw(type_kernel = "n", vec_data = x, n_pts = 100, seq_bws = NULL)
```

In Table 2 we enumerate the arguments and results of this function.

In `seq_bws`, by default, a sequence of 50 points is selected, from the range of the data divided by 200 to the range divided by 2. Next, the function CV in Equation 24 is evaluated on each of these points and then minimized.

In general, cross-validation functions in nonparametric bandwidth selection present several local minima. These minima are more likely to appear at too small values of the bandwidth (Hall and Marron 1991). In `CVbw`, the cross-validation function is evaluated in a sequence of

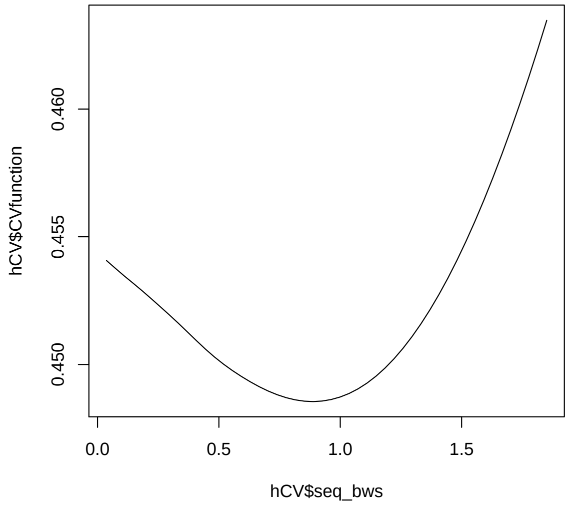

Figure 3: CV function obtained by `CVbw`.

values, beginning at a very small value and ending in a high one. The user can select the first and the last values and the size of the sequence to find the local minima, if there exist. In any case, the calculation of the minimum is made directly through the R function `which.min` that provides the index of the sequence minimizing the CV score. Therefore, it is very unlikely to obtain the minimum value in two different points.

In the following example, we compute the cross-validation bandwidth, using the Epanechnikov kernel, for a sample of 30 random numbers of a $N(0,1)$ distribution. The plot of the cross-validation function CV is shown in Figure 3.

```
R> x <- rnorm(30)
R> num_bws <- 50
R> seq_bws <- seq(((max(x)-min(x))/2)/50, (max(x)-min(x))/2,
+    length = num_bws)
R> hCV <- CVbw(type_kernel = "e", x, n_pts = 200, seq_bws)
R> hCV

[1] 0.8883513

R> plot(hCV$seq_bws, hCV$CVfunction, type = "l")
```

The next code lines allow us to compare the three bandwidth selection methods, using a normal kernel and a standard setting of parameters, in each case. In Figure 4 we can observe the results.

```
R> x <- rnorm(100)
R> h_CV <- CVbw(vec_data = x)$bw
R> h_AL <- ALbw(vec_data = x)
```

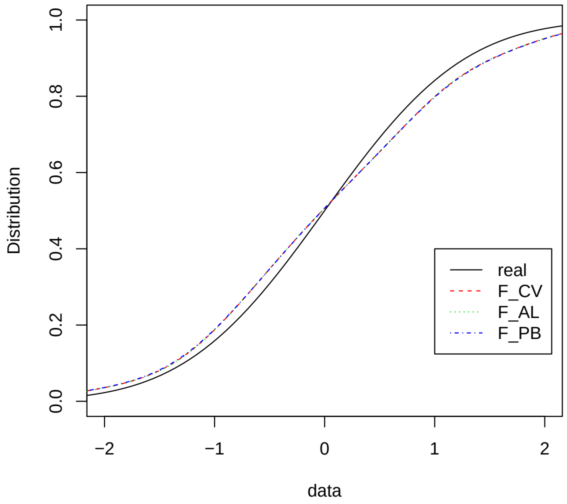

Figure 4: Distribution function estimation using the studied bandwidth selection methods.

```
R> h_PB <- PBbw(vec_data = x)
R> F_CV <- kde(vec_data = x, bw = h_CV)
R> F_AL <- kde(vec_data = x, bw = h_AL)
R> F_PB <- kde(vec_data = x, bw = h_PB)
R> y <- F_CV$grid
R> Ft <- pnorm(y)
R> plot(y, Ft, ylab = "Distribution", xlab = "data", type = "l",
+    xlim = c(-2, 2))
R> lines(y, F_CV$Estimated_values, type = "l", lty = 2, col = 2)
R> lines(y, F_AL$Estimated_values, type = "l", lty = 3, col = 3)
R> lines(y, F_PB$Estimated_values, type = "l", lty = 4, col = 4)
R> legend(1, 0.4, c("real", "F_CV", "F_AL", "F_PB"), lty = 1:4, col = 1:4)
```

4.2. Implementation of the associated functions

The functions of Equations 27, 29 and 31, defined in Section 3, have been programmed in `ef`, `mpr` and `rl`, respectively. Since the two first functions are directly defined from the distribution one, its calculation is straightforward (once we have an estimate of the mean rate λ). In the case of `rl`, because it is not possible to obtain a direct expression for the inverse of the distribution function estimator, we use a numerical technique for root finding. We have directly programmed the bisection method (dichotomy, Burden and Faires 2000), for its simplicity and speed (it has been used several times in similar problems; see e.g., del Río 2011). More details are given below.

Function `ef`. The exceedance probability of a concrete value `c` (a magnitude of a seismic event, a flow or pollutant level) will be exceeded in D time units. From a data set `vec_data`, the function is invoked in the form:

```
ef(type_kernel = "n", vec_data, c,
  bw = PBbw(type_kernel = "n", vec_data, 2),
  Dmin = 0, Dmax = 15, size_grid = 50, lambda)
```

Choose chooses a part choses a where, as before, four kernel types are allowed. `c` is the target in which we calculate the risk function, which is evaluated on a grid of 50 (by default) points from `Dmin` to `Dmax`. The mean activity rate `lambda` is the parameter of the Poisson variable that controls the occurrence process of the events (earthquakes, winds). `ef` chooses the plug-in bandwidth of Polansky and Baker to select the bandwidth. The output is a list containing the used grid, the bandwidth and the estimated values on the grid. An example is shown in Section 5.1.

Function `mpr`. Function that estimates the mean period of time between two events with the same concrete level (magnitude of an earthquake, flow level, concentration of a pollutant, wind speed). From a data set `vec_data`, the function is invoked in the form:

```
mrp(type_kernel = "n", vec_data, y = NULL,
  bw = PBbw(type_kernel = "n", vec_data, 2), lambda)
```

The plug-in bandwidth of Polansky and Baker is calculated from the data set `vec_data`. For a grid defined as a sequence of 50 points between the minimum and the maximum of the data, the output allows to plot directly the resulting function, because a list containing the grid used, the bandwidth and the estimated values is obtained. As for the above case, a real application can be seen in Section 5.1.

Function `rl`. This function is directly related with the last one, but with easier handling.

```
rl(type_kernel = "n", vec_data, T,
  bw = PBbw(type_kernel = "n", vec_data, 2))
```

A detailed description, through an example using a hydrological application, is given in Section 5.2.

5. Data description and applications

5.1. Northwest of the Iberian Peninsula data

The data of `nwip` correspond to the earthquakes occurring in the northwest of the Iberian Peninsula, from 1924-11-25 to 2010-07-31. The coordinates of the zone are 41 N – 44 N and 6 W – 10 W, and involve the autonomic region of Galicia (Spain) and northern Portugal. The data can be obtained from the National Geographic Institute (IGN) of Spain, through the web page http://www.ign.es/.

The epicenters of the earthquakes are marked in a map in Figure 5, done with MATLAB (The MathWorks, Inc. 2010) for its better resolution than an R map for this area. There are 3491 earthquakes. A geophysical description of the characteristics of this area can be seen in Rueda and Mezcua (2001), Martínez-Díaz, Capote, Tsige, Martin-Gonzalez, Villamor, and Insua (2006) and Díaz, Gallart, Gaspa, Ruiz, and Córdoba (2008). A nonparametric study of

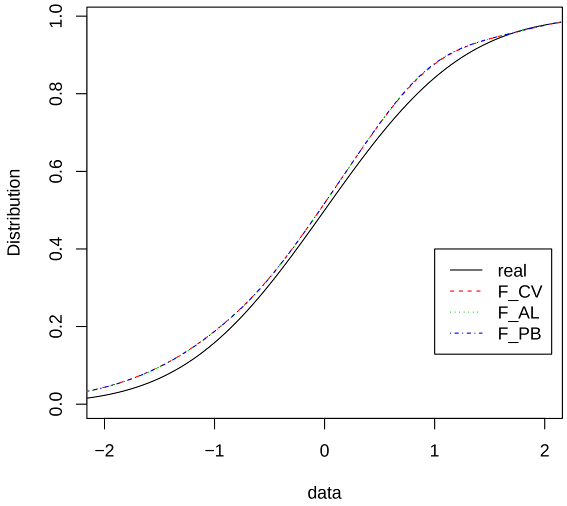

Figure 5: Epicenters of earthquakes included in the data set `nwip`.

the hazard function of a subset of these data is made in Estévez-Pérez, Lorenzo-Cimadevila, and del Río (2002).

In the following example, we need to load the packages **chron** (James and Hornik 2011) and **date** (Therneau, Lumley, Halvorsen, and Hornik 2011) to work with the dates of the earthquake data and compute the number of years. We select those earthquakes with magnitude greater than 3.0, because the set of data with magnitude bigger than 3.0 fulfills the Guttenberg-Richter law. That is, 3.0 is the completeness magnitude for this catalogue; (see e.g., Woessner and Wiemer 2005).

```
R> library("chron")
R> library("date")
R> data("nwip")
R> mg <- nwip$magnitude[nwip$magnitude > 3.0]
R> x1 <- nwip$year
R> x2 <- nwip$month
R> x3 <- nwip$day
R> ys <- paste(x1,x2,x3)
R> earthquake_date <- as.character(ys)
R> y1s <- as.date(earthquake_date, order = "ymd")
R> y2s <- as.POSIXct(y1s)
R> z <- years(y2s)
R> n.years <- length(levels(z))
R> lambda <- length(mg)/n.years
```

The value `lambda` is the mean earthquake rate per year, that is, we assume that the occurrence process of the earthquakes is a Poisson variable with parameter `lambda`. With the following code we can show, in Figure 6, the exceedance function for the values of the magnitude equal

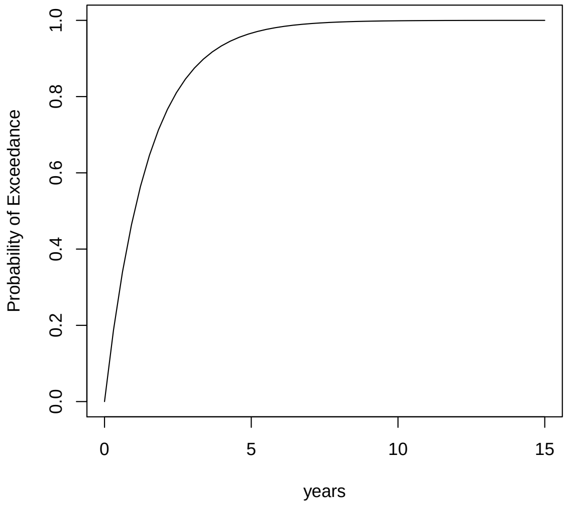

Figure 6: Exceedance function estimation for values of magnitude equal to 4 using the function `ef`.

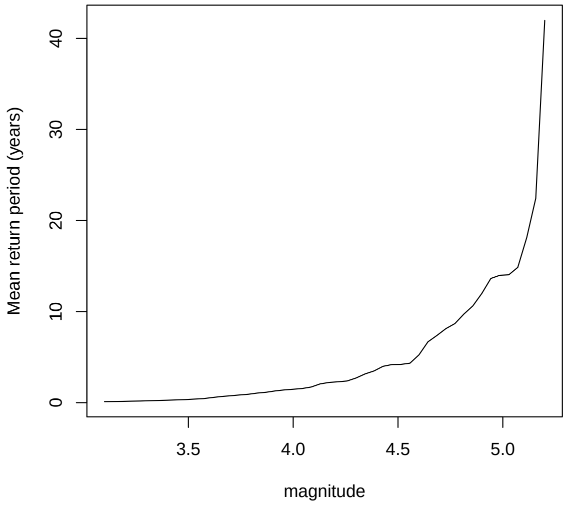

Figure 7: Mean return period estimation for earthquakes of the same magnitude using the function `mpr`.

to 4. We observe that the risk function of occurrence of an earthquake with magnitude 4 or bigger is higher when the number of years increases, being the probability practically equal to 1 when the number of years approaches 10.

```
R> est <- ef(vec_data = mg, c = 4, lambda = lambda)
R> plot(est$grid, est$Estimated_values, type = "l", xlab = "years",
+    ylab = "Probability of Exceedance")
```

Now, we plot in Figure 7 the mean return period between earthquakes of the same magnitude, using the function mpr. As expected, the bigger the magnitude, the return period increases, and for very high values, the number of years tends to infinity (the maximum magnitude of the data set is 5.2).

```
R> est2 <- mrp(vec_data = mg, lambda =lambda)
R> plot(est2$grid, est2$Estimated_values, type = "l", xlab = "magnitude",
+    ylab = "Mean return period(years)")
```

5.2. Saltriver data

The file saltriver contains the annual peak instantaneous flow levels of the Salt River near Roosevelt, AZ, USA, for the period 1924-2009, obtained from the National Water Information System at http://waterdata.usgs.gov/nwis. This data set appears in the R package **extRemes** (Gilleland and Katz 2011), but limited to 1999. These data (in cumecs; $m^3 s^{-1}$) were exhaustively analyzed in Katz *et al.* (2002), where the authors fitted a generalized extreme value (GEV) distribution Coles (2001) in a classical (parametric) extreme values analysis to obtain the functions of Equation 25, Equation 28 and Equation 30. A graph of the relevant variable appears in Figure 8, as a result of the code

```
R> data("saltriver")
R> peak <- saltriver$peakflow
R> year <- saltriver$year
R> plot(year, peak, type = "l", ylab = "Annual peak flow")
```

Now, we make a comparison between parametric and nonparametric estimates for computing the mean return period. Firstly, we estimate nonparametrically the mean return period. Next,

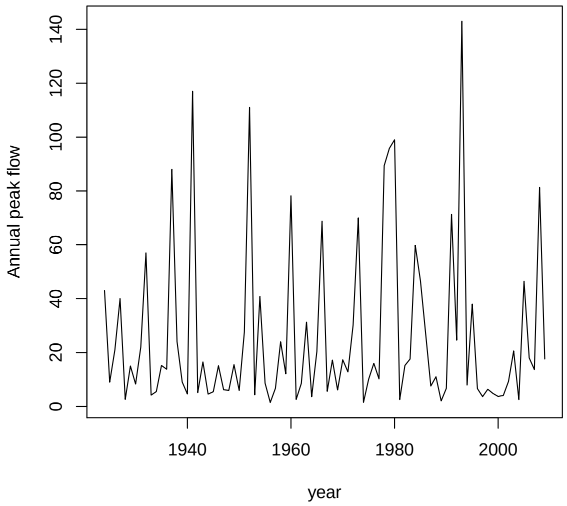

Figure 8: Annual peak flow levels for data in saltriver.

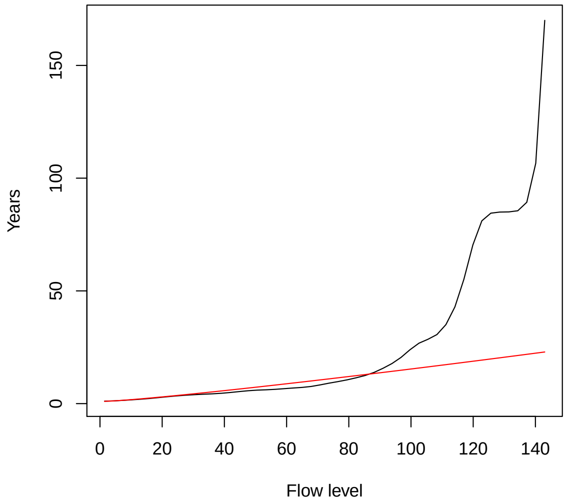

Figure 9: Mean return period estimated parametrically (red line) and nonparametrically (black line).

we fit a GEV distribution by means of the R package **evir** (Pfaff, McNeil, and Stephenson 2012). This distribution depends on three parameters, and its estimated values are $(\hat{\mu}, \hat{\sigma}, \hat{\gamma})$ = (0.87, 8.17, 8.33).

```
R> library("evir")
R> rp <- mrp(type_kernel = "n", vec_data = peak, lambda = 1)
R> plot(rp$grid, rp$Estimated_values, type = "l", xlab = "Flow level",
+    ylab = "Years")
R> x1 <- gev(peak)
R> loc1 <- x1$par.ests[1]
R> scale1 <- x1$par.ests[2]
R> shape1 <- x1$par.ests[3]
R> lines(rp$grid, 1/(1 - pgev(rp$grid, loc1, scale1, shape1)),
+    type = "l", col = 2)
```

In Figure 9 we plot the mean return period estimated by the resulting estimators. The return period estimated by the fitted GEV (`gev`) distribution (red line) is far from the nonparametric estimator, revealing that this would not be a good theoretical distribution for these data, at least for high values of the flow level (del Río 2011).

Finally, we compute the return levels for a period of 2 to 100 years, and show them in Figure 10.

The analysis of Figure 10 reveals that, for example, in 20 years, we can expect that the flow level of 100 cms will be exceeded at least once. An interesting jump can be seen at the levels 120-140, which will be exceeded after 80 years. The code for this purpose appears below.

```
R> T <- seq(2, 100, length.out = 100)
```

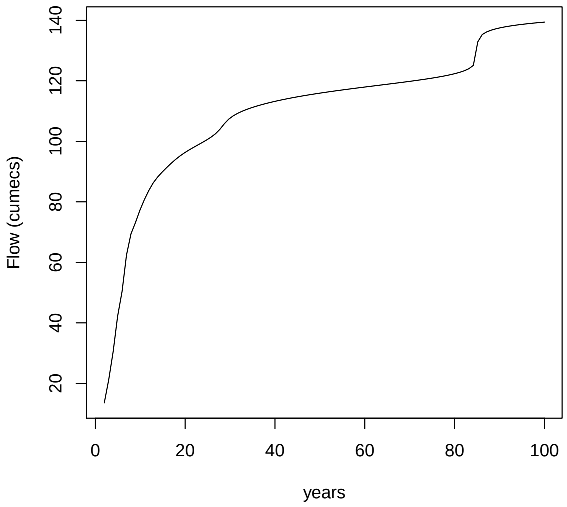

Figure 10: Return levels estimation for a period from 2 to 100 years.

```
R> ret.lev <- rl(vec_data = peak, T = T)
R> plot(T, ret.lev, type = "l", xlab = "years", ylab = "Flow (cumecs)")
```

6. Conclusions

This work discusses the package **kerdiest** and its applications to real problems. The nonparametric function distribution estimation appears as a powerful tool to treat several real questions related with the risk function. In this setting, computation of concerning parameters such as the return levels or mean return periods are valuable allies in the study of nature hazards.

To estimate the distribution function using a kernel estimator it is necessary to have a bandwidth parameter selection procedure. In the specialized literature, four methods have been developed in this direction, two of the plug-in type and two using cross-validation ideas. Because one of the last two methods did not behave well in simulation studies (Altman and Leger 1995), we have only implemented the modified cross-validation method of Bowman *et al.* (1998). Moreover, we have complemented these routines with the programming of the mentioned associated functions for real data applications, where we have directly made use of the fastest bandwidth selection method. We think that this package can be of interest to nonparametric practitioners of different scientific fields.

Acknowledgments

We would like to thank the referees and the Associate Editor for a number of comments which have lead to an improved presentation of the paper.

References

Altman N, Leger C (1995). "Bandwidth Selection for Kernel Distribution Function Estimation." *Journal of Statistical Planning and Inference*, **46**, 195–214.

Bowman A, Hall P, Prvan T (1998). "Bandwidth Selection for the Smoothing of Distribution Functions." *Biometrika*, **85**, 799–808.

Burden RL, Faires JD (2000). *Numerical Analysis*. Brooks/Cole.

Coles SC (2001). *An Introduction to Statistical Modeling of Extreme Values*. Springer-Verlag, London.

del Río AQ (1996). "Comparison of Bandwidth Selectors in Nonparametric Regression under Dependence." *Computational Statistics & Data Analysis*, **21**, 563–580.

del Río AQ (2011). "On Bandwidth Selection for Nonparametric Estimation in Flood Frequency Analysis." *Hydrological Processes*, **25**, 671–678.

del Río AQ, Francisco-Fernández F (2011). "Nonparametric Functional Data Estimation Applied to Ozone Data: Prediction and Extreme Value Analysis." *Chemosphere*, **82**, 800–808.

Díaz J, Gallart J, Gaspa O, Ruiz M, Córdoba D (2008). "Seismicity Analysis at the Prestige Oil-Tanker Wreck Area (Galicia Margin, NW of Iberia)." *Marine Geology*, **249**, 150–165.

Elsner JB, Jagger TH, Tsonis AA (2006). "Estimated Return Periods for Hurricane Katrina." *Geophysical Research Letters*, **33**, L08704.

Estévez-Pérez MG, Lorenzo-Cimadevila H, del Río AQ (2002). "Nonparametric Analysis of the Time Structure of Seismicity in a Geographic Region." *Annals of Geophysics*, **45**, 497–511.

Gilleland E, Katz RW (2011). "New Software to Analyze How Extremes Change over Time." *Eos*, **92**(2), 13–14.

Gomes L, Arrúe JL, López MV, Sterk G, Richard D (2003). "Wind Erosion in a Semiarid Agricultural Area of Spain: The WELSONS Project." *Catena*, **52**, 235–256.

Hall P, Marron JS (1991). "Local Minima in Cross-Validation Functions." *Journal of the Royal Statistical Society B*, **53**, 245–252.

Hill PD (1985). "Kernel Estimation of a Distribution Function." *Communications in Statistics, Theory and Methods*, **14**, 605–620.

James D, Hornik K (2011). ***chron**: Chronological Objects which Can Handle Dates and Times*. R package version 2.3-42, URL http://CRAN.R-project.org/package=chron.

Jones MC, Marron JS, Sheather SJ (1996). "A Brief Survey of Bandwidth Selection for Density Estimation." *Journal of the American Statistical Society*, **91**, 401–407.

Katz RW, Parlange MB, Naveau P (2002). "Statistics of Extremes in Hydrology." *Advances in Water Resources*, **25**, 1287–1304.

Küchenhoff H, Thamerus M (1996). "Extreme Value Analysis of Munich Air Pollution Data." *Environmental and Ecological Statistics*, **3**, 127–141.

Martínez-Díaz JJ, Capote R, Tsige M, Martin-Gonzalez F, Villamor P, Insua JM (2006). "Seismic Triggering in a Stable Continental Area: The Lugo 1995–1997 Seismic Sequences (NW Spain)." *Journal of Geodynamics*, **41**, 440–449.

Nadaraya EA (1964). "On Estimating Regression." *Theory of Probability and Applications*, **10**, 186–90.

Orlecka-Sikora B (2008). "Resampling Methods for Evaluating the Uncertainty of the Nonarametric Magnitude Distribution Estimation in the Probabilistic Seismic Hazard Analysis." *Tectonophysics*, **456**, 38–51.

Parzen E (1962). "On Estimation of a Probability Density Function and Mode." *The Annals of Mathematical Statistics*, **32**, 1065–1076.

Pfaff B, McNeil A, Stephenson A (2012). ***evir**: Extreme Values in R*. R package version 1.7-3, URL http://CRAN.R-project.org/package=evir.

Polanski A, Baker ER (2000). "Multistage Plug-in Bandwidth Selection for Kernel Distribution Function Estimates." *Journal of Statistical Computation and Simulation*, **65**, 63–80.

R Development Core Team (2012). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Reiss RD (1981). "Nonparametric Estimation of Smooth Distribution Functions." *Scandinavian Journal of Statistics*, **8**, 116–119.

Rueda J, Mezcua J (2001). *Sismicidad, Sismotectónica y Peligrosidad Sísmica en Galicia*. 35. IGN Technical Publication, Spain.

Sarda P (1993). "Smoothing Parameter Selection for Smooth Distribution Function." *Journal of Statistical Planning and Inference*, **35**, 65–75.

Scheitlin KN, Elsner JB, Malmstadt JC, Hodges RE, Jagger TH (2010). "Toward Increased Utilization of Historical Hurricane Chronologies." *Journal of Geophysical Research*, **115**, D03108.

Silverman BW (1986). *Density Estimation for Statistics and Data Analysis*. Chapman & Hall, London.

The MathWorks, Inc (2010). *MATLAB – The Language of Technical Computing, Version 7.10*. The MathWorks, Inc., Natick, Massachusetts. URL http://www.mathworks.com/products/matlab/.

Therneau T, Lumley T, Halvorsen K, Hornik K (2011). ***date**: Functions for Handling Dates*. R package version 1.2-32, URL http://CRAN.R-project.org/package=date.

Wand MP, Jones MC (1995). *Kernel Smoothing*. Chapman & Hall, London.

Woessner J, Wiemer S (2005). "Assessing the Quality of Earthquake Catalogues: Estimating the Magnitude of Completeness and Its Uncertainty." *Bulletin of the Seismological Society of America*, **95**, 684–698.

Affiliation:

Alejandro Quintela-del-Río
Department of Mathematics
University of A Coruña
15071 A Coruña, Spain
E-mail: aquintela@udc.es

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume 50, Issue 8 *Submitted:* 2011-04-07
August 2012 *Accepted:* 2012-06-18